              Use of GOST Signature Algorithms in DNSKEY
                and RRSIG Resource Records for DNSSEC

Abstract

   This document describes how to produce digital signatures and hash
   functions using the GOST R 34.10-2001 and GOST R 34.11-94 algorithms
   for DNSKEY, RRSIG, and DS resource records, for use in the Domain
   Name System Security Extensions (DNSSEC).

Status of This Memo

   This is an Internet Standards Track document.

   This document is a product of the Internet Engineering Task Force
   (IETF). It represents the consensus of the IETF community. It has
   received public review and has been approved for publication by the
   Internet Engineering Steering Group (IESG). Further information on
   Internet Standards is available in Section 2 of RFC 5741.

   Information about the current status of this document, any errata,
   and how to provide feedback on it may be obtained at
   http://www.rfc-editor.org/info/rfc5933.

Table of Contents

1. Introduction

   The Domain Name System (DNS) is the global hierarchical distributed
   database for Internet Naming. The DNS has been extended to use
   cryptographic keys and digital signatures for the verification of the
   authenticity and integrity of its data. RFC 4033 [RFC4033], RFC 4034
   [RFC4034], and RFC 4035 [RFC4035] describe these DNS Security
   Extensions, called DNSSEC.

   RFC 4034 describes how to store DNSKEY and RRSIG resource records,
   and specifies a list of cryptographic algorithms to use. This
   document extends that list with the signature and hash algorithms
   GOST R 34.10-2001 ([GOST3410], [RFC5832]) and GOST R 34.11-94
   ([GOST3411], [RFC5831]), and specifies how to store DNSKEY data and
   how to produce RRSIG resource records with these algorithms.

   Familiarity with DNSSEC and with GOST signature and hash algorithms
   is assumed in this document.

   The term "GOST" is not officially defined, but is usually used to
   refer to the collection of the Russian cryptographic algorithms
   GOST R 34.10-2001 [RFC5832], GOST R 34.11-94 [RFC5831], and

GOST 28147-89 [RFC5830]. Since GOST 28147-89 is not used in DNSSEC,
"GOST" will only refer to GOST R 34.10-2001 and GOST R 34.11-94 in
this document.

1.1. Terminology

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT",
"SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this
document are to be interpreted as described in [RFC2119].

2. DNSKEY Resource Records

The format of the DNSKEY RR can be found in RFC 4034 [RFC4034].

GOST R 34.10-2001 public keys are stored with the algorithm
number 12.

The wire format of the public key is compatible with RFC 4491
[RFC4491]:

According to [GOST3410] and [RFC5832], a public key is a point on the
elliptic curve $Q = (x,y)$.

The wire representation of a public key MUST contain 64 octets, where
the first 32 octets contain the little-endian representation of x and
the second 32 octets contain the little-endian representation of y.

Corresponding public key parameters are those identified by
id-GostR3410-2001-CryptoPro-A-ParamSet (1.2.643.2.2.35.1) [RFC4357],
and the digest parameters are those identified by
id-GostR3411-94-CryptoProParamSet (1.2.643.2.2.30.1) [RFC4357].

2.1. Using a Public Key with Existing Cryptographic Libraries

At the time of this writing, existing GOST-aware cryptographic
libraries are capable of reading GOST public keys via a generic X509
API if the key is encoded according to RFC 4491 [RFC4491],
Section 2.3.2.

To make this encoding from the wire format of a GOST public key with
the parameters used in this document, prepend the 64 octets of key
data with the following 37-byte sequence:

    0x30 0x63 0x30 0x1c 0x06 0x06 0x2a 0x85 0x03 0x02 0x02 0x13 0x30
    0x12 0x06 0x07 0x2a 0x85 0x03 0x02 0x02 0x23 0x01 0x06 0x07 0x2a
    0x85 0x03 0x02 0x02 0x1e 0x01 0x03 0x43 0x00 0x04 0x40

2.2. GOST DNSKEY RR Example

   Given a private key with the following value (the value of the
   GostAsn1 field is split here into two lines to simplify reading; in
   the private key file, it must be in one line):

      Private-key-format: v1.2
      Algorithm: 12 (ECC-GOST)
      GostAsn1: MEUCAQAwHAYGKoUDAgITMBIGByqFAwICIwEGByqFAwICHgEEIgQg/9M
              iXtXKg9FDXDN/R9CmVhJDyuzRAIgh4tPwCu4NHIs=

   The following DNSKEY RR stores a DNS zone key for example.net:

      example.net. 86400 IN DNSKEY 256 3 12 (
                              aRS/DcPWGQj2wVJydT8EcAVoC0kXn5pDVm2I
                              MvDDPXeD32dsSKcmq8KNVzigjL4OXZTV+t/6
                              w4X1gpNrZiC01g==
) ; key id = 59732

3. RRSIG Resource Records

   The value of the signature field in the RRSIG RR follows RFC 4490
   [RFC4490] and is calculated as follows. The values for the RDATA
   fields that precede the signature data are specified in RFC 4034
   [RFC4034].

   hash = GOSTR3411(data)

   where "data" is the wire format data of the resource record set that
   is signed, as specified in RFC 4034 [RFC4034].

   The hash MUST be calculated with GOST R 34.11-94 parameters
   identified by id-GostR3411-94-CryptoProParamSet [RFC4357].

   The signature is calculated from the hash according to the
   GOST R 34.10-2001 standard, and its wire format is compatible with
   RFC 4490 [RFC4490].

   Quoting RFC 4490:

   "The signature algorithm GOST R 34.10-2001 generates a digital
   signature in the form of two 256-bit numbers, r and s. Its octet
   string representation consists of 64 octets, where the first
   32 octets contain the big-endian representation of s and the second
   32 octets contain the big-endian representation of r".

3.1. RRSIG RR Example

   With the private key from Section 2.2, sign the following RRSet,
   consisting of one A record:

      www.example.net. 3600 IN A 192.0.2.1

   Setting the inception date to 2000-01-01 00:00:00 UTC and the
   expiration date to 2030-01-01 00:00:00 UTC, the following signature
   RR will be valid:

     www.example.net. 3600 IN RRSIG A 12 3 3600 20300101000000 (
                                    20000101000000 59732 example.net.
                                    7vzzz6iLOmvtjs5FjVjSHT8XnRKFY15ki6Kp
                                    kNPkUnS8iIns0Kv4APT+D9ibmHhGri6Sfbyy
                                    zi67+wBbbW/jrA==)

   Note: The ECC-GOST signature algorithm uses random data, so the
   actual computed signature value will differ between signature
   calculations.

4. DS Resource Records

   The GOST R 34.11-94 digest algorithm is denoted in DS RRs by the
   digest type 3. The wire format of a digest value is compatible with
   RFC 4490 [RFC4490], that is, the digest is in little-endian
   representation.

   The digest MUST always be calculated with GOST R 34.11-94 parameters
   identified by id-GostR3411-94-CryptoProParamSet [RFC4357].

4.1. DS RR Example

   For Key Signing Key (KSK):

      example.net. 86400   DNSKEY 257 3 12 (
                                   LMgXRHzSbIJGn6i16K+sDjaDf/k1o9DbxScO
                                   gEYqYS/rlh2Mf+BRAY3QHPbwoPh2fkDKBroF
                                   SRGR7ZYcx+YIQw==
) ; key id = 40692

   The DS RR will be

      example.net. 3600 IN DS 40692 12 3 (
              22261A8B0E0D799183E35E24E2AD6BB58533CBA7E3B14D659E9CA09B
              2071398F)

5. Deployment Considerations

5.1. Key Sizes

   According to RFC 4357 [RFC4357], the key size of GOST public keys
   MUST be 512 bits.

5.2. Signature Sizes

   According to the GOST R 34.10-2001 digital signature algorithm
   specification ([GOST3410], [RFC5832]), the size of a GOST signature
   is 512 bits.

5.3. Digest Sizes

   According to GOST R 34.11-94 ([GOST3411], [RFC5831]), the size of a
   GOST digest is 256 bits.

6. Implementation Considerations

6.1. Support for GOST Signatures

   DNSSEC-aware implementations MAY be able to support RRSIG and DNSKEY
   resource records created with the GOST algorithms as defined in this
   document.

6.2. Support for NSEC3 Denial of Existence

   Any DNSSEC-GOST implementation MUST support both NSEC [RFC4035] and
   NSEC3 [RFC5155].

7. Security Considerations

   Currently, the cryptographic resistance of the GOST R 34.10-2001
   digital signature algorithm is estimated as $2**128$ operations of
   multiple elliptic curve point computations on prime modulus of order
   $2**256$.

   Currently, the cryptographic resistance of the GOST R 34.11-94 hash
   algorithm is estimated as $2**128$ operations of computations of a step
   hash function. (There is a known method to reduce this estimate to
   $2**105$ operations, but it demands padding the colliding message with
   1024 random bit blocks each of 256-bit length; thus, it cannot be
   used in any practical implementation).

8. IANA Considerations

   This document updates the IANA registry "DNS Security Algorithm
   Numbers" [RFC4034]. The following entries have been added to the
   registry:

```
                                  Zone   Trans.
   Value  Algorithm        Mnemonic Signing Sec.   References   Status
    12     GOST R 34.10-2001 ECC-GOST    Y    *     RFC 5933   OPTIONAL
```

   This document updates the RFC 4034 Digest Types assignment
   ([RFC4034], Section A.2) by adding the value and status for the
   GOST R 34.11-94 algorithm:

```
     Value    Algorithm       Status
       3     GOST R 34.11-94  OPTIONAL
```

9. Acknowledgments

   This document is a minor extension to RFC 4034 [RFC4034]. Also, we
   tried to follow the documents RFC 3110 [RFC3110], RFC 4509 [RFC4509],
   and RFC 4357 [RFC4357] for consistency. The authors of and
   contributors to these documents are gratefully acknowledged for their
   hard work.

   The following people provided additional feedback, text, and valuable
   assistance: Dmitry Burkov, Jaap Akkerhuis, Olafur Gundmundsson,
   Jelte Jansen, and Wouter Wijngaards.

10. References

10.1. Normative References

   [GOST3410] "Information technology. Cryptographic data security.
               Signature and verification processes of [electronic]
               digital signature.", GOST R 34.10-2001, Gosudarstvennyi
               Standard of Russian Federation, Government Committee of
               Russia for Standards, 2001. (In Russian).

   [GOST3411] "Information technology. Cryptographic data security.
               Hashing function.", GOST R 34.11-94, Gosudarstvennyi
               Standard of Russian Federation, Government Committee of
               Russia for Standards, 1994. (In Russian).

   [RFC2119]   Bradner, S., "Key words for use in RFCs to Indicate
               Requirement Levels", BCP 14, RFC 2119, March 1997.

   [RFC3110] Eastlake 3rd, D., "RSA/SHA-1 SIGs and RSA KEYs in the
              Domain Name System (DNS)", RFC 3110, May 2001.

   [RFC4033] Arends, R., Austein, R., Larson, M., Massey, D., and S.
              Rose, "DNS Security Introduction and Requirements",
              RFC 4033, March 2005.

   [RFC4034] Arends, R., Austein, R., Larson, M., Massey, D., and S.
              Rose, "Resource Records for the DNS Security Extensions",
              RFC 4034, March 2005.

   [RFC4035] Arends, R., Austein, R., Larson, M., Massey, D., and S.
              Rose, "Protocol Modifications for the DNS Security
              Extensions", RFC 4035, March 2005.

   [RFC4357] Popov, V., Kurepkin, I., and S. Leontiev, "Additional
              Cryptographic Algorithms for Use with GOST 28147-89,
              GOST R 34.10-94, GOST R 34.10-2001, and GOST R 34.11-94
              Algorithms", RFC 4357, January 2006.

   [RFC4490] Leontiev, S., Ed. and G. Chudov, Ed., "Using the
              GOST 28147-89, GOST R 34.11-94, GOST R 34.10-94, and
              GOST R 34.10-2001 Algorithms with Cryptographic Message
              Syntax (CMS)", RFC 4490, May 2006.

   [RFC4491] Leontiev, S., Ed. and D. Shefanovski, Ed., "Using the
              GOST R 34.10-94, GOST R 34.10-2001, and GOST R 34.11-94
              Algorithms with the Internet X.509 Public Key
              Infrastructure Certificate and CRL Profile", RFC 4491,
              May 2006.

   [RFC5155] Laurie, B., Sisson, G., Arends, R., and D. Blacka, "DNS
              Security (DNSSEC) Hashed Authenticated Denial of
              Existence", RFC 5155, March 2008.

10.2. Informative References

   [RFC4509] Hardaker, W., "Use of SHA-256 in DNSSEC Delegation Signer
              (DS) Resource Records (RRs)", RFC 4509, May 2006.

   [RFC5830] Dolmatov, V., Ed., "GOST 28147-89: Encryption,
              Decryption, and Message Authentication Code (MAC)
              Algorithms", RFC 5830, March 2010.

   [RFC5831] Dolmatov, V., Ed., "GOST R 34.11-94: Hash Function
              Algorithm", RFC 5831, March 2010.

   [RFC5832]   Dolmatov, V., Ed., "GOST R 34.10-2001: Digital Signature
                Algorithm", RFC 5832, March 2010.

Authors' Addresses

    Vasily Dolmatov (editor)
    Cryptocom Ltd.
    14/2, Kedrova St.
    Moscow,   117218
    Russian Federation

    Phone: +7 499 124 6226
    EMail: dol@cryptocom.ru

    Artem Chuprina
    Cryptocom Ltd.
    14/2, Kedrova St.
    Moscow,   117218
    Russian Federation

    Phone: +7 499 124 6226
    EMail: ran@cryptocom.ru

    Igor Ustinov
    Cryptocom Ltd.
    14/2, Kedrova St.
    Moscow,   117218
    Russian Federation

    Phone: +7 499 124 6226
    EMail: igus@cryptocom.ru